

[illegible]

3

Sy

MT

MT
MT

MT

MT

MT
MTMT
MTMT
MTMT
MT

MT
MT

MT
MT

MT

MT

MT
MT

MI
MT

MT
MT

MT
MT

MT
MT

MT
MT

MT

111

227

MT

MT
MT

MT

MT

M1
M1MT
MT

M1

M1

10

1

10

100

10

1

```

LL               IIIIII               SSSSSSSS
LL               IIIIII               SSSSSSSS
LL               II                   SS
LL               II                   SS
LL               II                   SS
LL               II                   SS
LL               II                   SSSSSS
LL               II                   SSSSSS
LL               II                   SS
LL               II                   SS
LL               II                   SS
LL               II                   SS
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS

```

(2)	54
(3)	69
(4)	113
(5)	176
(6)	267
(7)	330

HISTORY ; Detailed Current Edit History
DECLARATIONS ; Declarative Part of Module
MTH\$GACOS - Standard G Floating GACOS
MTH\$GACOS_R7 - Special GACOS routine
MTH\$GACOSD - Standard G Floating GACOSD
MTH\$GACOSD_R7 - Special GACOSD routine


```
0000 1 .TITLE MTH$GACOS ; G Floating Point Arc-cosine routine
0000 2 ; (GACOS,GACOSD)
0000 3 .IDENT /1-006/ ; File: MTHGACOS.MAR EDIT: RNH1006
0000 4 :
0000 5 :*****
0000 6 :*
0000 7 :* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 :* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 :* ALL RIGHTS RESERVED.
0000 10 :*
0000 11 :* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 :* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 :* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 :* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 :* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 :* TRANSFERRED.
0000 17 :*
0000 18 :* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 :* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 :* CORPORATION.
0000 21 :*
0000 22 :* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 :* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 :*
0000 25 :*
0000 26 :*****
0000 27 :
0000 28 :
0000 29 : FACILITY: MATH LIBRARY
0000 30 : ++
0000 31 : ABSTRACT:
0000 32 :
0000 33 : MTH$GACOS is a function which returns the G floating point arc-cosine
0000 34 : in radians of its G floating point argument. The call is standard
0000 35 : call-by-reference.
0000 36 :
0000 37 : MTH$GACOSD is a function which returns the G floating point arc-cosine
0000 38 : in degrees of its G floating point argument. The call is standard
0000 39 : call-by-reference.
0000 40 :
0000 41 : --
0000 42 :
0000 43 : VERSION: 1
0000 44 :
0000 45 : HISTORY:
0000 46 : AUTHOR:
0000 47 : Steven B. Lionel, 15-Jan-79: Version 1
0000 48 :
0000 49 : MODIFIED BY:
0000 50 :
0000 51 :
0000 52 :
```

MTH\$GACOS
1-006

N 4
; G Floating Point Arc-cosine routine 16-SEP-1984 01:24:21 VAX/VMS Macro V04-00
HISTORY ; Detailed Current Edit History 6-SEP-1984 11:23:16 [MTHRTL.SRC]MTHGACOS.MAR;1 Page 2
(2)

```
0000 54 .SBTTL HISTORY ; Detailed Current Edit History
0000 55
0000 56
0000 57 ; ALGORITHMIC DIFFERENCES FROM FP-11/C ROUTINE: none
0000 58 ;
0000 59 ; Edit History for Version 1 of MTH$GACOS
0000 60 ;
0000 61 ; 1-001 - Adapted from MTH$DACOS version 1-001. SBL 15-Jan-79
0000 62 ; 1-002 - Change JSB entry to MTH$GACOS R7. RBG 28-Sept-1979
0000 63 ; 1-003 - Added degree entry points. RNH 22-MAR-1981
0000 64 ; 1-004 - Modified computation of  $1 - X^2$  to avoid loss of significance
0000 65 ; for  $|X| \geq 1/2$ . RNH 02-Sept-1981
0000 66 ; 1-005 - Changed shared external references to G^. RNH 02-Oct-81
0000 67 ; 1-006 - Eliminated symbolic short literals. RNH 15-Oct-81
```



```
0000 69      .SBTTL DECLARATIONS      ; Declarative Part of Module
0000 70
0000 71      ;
0000 72      ; INCLUDE FILES:          OTSPARAMS.MAR
0000 73      ;
0000 74      ; EXTERNAL SYMBOLS:
0000 75      ;
0000 76
0000 77      .DSABL GBL                  ; force undefineds to error
0000 78      .EXTRN MTH$GATAN R7         ; arctangent routine
0000 79      .EXTRN MTH$GATAN R7         ; arctangent routine
0000 80      .EXTRN MTH$GSQRT R5         ; square root routine
0000 81      .EXTRN MTH$$SIGNAL          ; math signal routine
0000 82      .EXTRN MTH$K_INVARGMAT      ; Error code
0000 83
0000 84      ; EQUATED SYMBOLS:
0000 85
00000004 0000 86      value = 4          ; value.rg.r
0000 87
0000 88      ;
0000 89      ; MACROS:                  none
0000 90      ;
0000 91      ; PSECT DECLARATIONS:
0000 92
00000000 0000 93      .PSECT _MTH$CODE      PIC,SHR, LONG, EXE, NOWRT
0000 94      ; program section for math routines
0000 95      ;
0000 96      ; OWN STORAGE:             none
0000 97      ;
0000 98      ;
0000 99      ; CONSTANTS:
0000 100     ;
0000 101     ;
0000 102     G_PI_OVER_2:
2D18 5444 21FB 4019 0000 103     .WORD ^0040031, ^0020773, ^0052104, ^0026430
0008 104     ; PI/2
0008 105     G_PI:
2D18 5444 21FB 4029 0008 106     .WORD ^0040051, ^0020773, ^0052104, ^0026430
0010 107     ; PI
0010 108     G_90:
00000000 80004076 0010 109     .LONG ^X80004076, ^X0          ; 90
0018 110     G_180:
00000000 80004086 0018 111     .LONG ^X80004086, ^X0          ; 180
```

```
0020 113 .SBTTL MTH$GACOS - Standard G Floating GACOS
0020 114
0020 115
0020 116 :++
0020 117 : FUNCTIONAL DESCRIPTION:
0020 118
0020 119 : GACOS - G floating point function
0020 120
0020 121 : GACOS(X) is computed as:
0020 122
0020 123 : If X = 0, then GACOS(X) = PI/2.
0020 124 : If X = 1, then GACOS(X) = 0.
0020 125 : If X = -1, then GACOS(X) = PI.
0020 126 : If 0 < X < 1/2, then GACOS(X) = ATAN(SQRT(1-X**2)/X).
0020 127 : If 1/2 <= X < 1, then GACOS(X) = ATAN(SQRT((1-X)*(1+X))/X).
0020 128 : If -1/2 < X < 0, then GACOS(X) = ATAN(SQRT(1-X**2)/X) + PI.
0020 129 : If -1 < X <= -1/2, then GACOS(X) = ATAN(SQRT((1-X)*(1+X))/X) + PI
0020 130 : If 1 < |X|, error.
0020 131
0020 132 : CALLING SEQUENCE:
0020 133
0020 134 : gacos.wg.v = MTH$GACOS(x.rg.r)
0020 135
0020 136 : INPUT PARAMETERS:
0020 137
00000004 0020 138 : LONG = 4 ; define longword multiplier
00000004 0020 139 : x = 1 * LONG ; Contents of x is the argument
0020 140
0020 141 : IMPLICIT INPUTS: none
0020 142
0020 143 : OUTPUT PARAMETERS:
0020 144
0020 145 : VALUE: G floating arc-cosine of the argument
0020 146
0020 147 : IMPLICIT OUTPUTS: none
0020 148
0020 149 : COMPLETION CODES: none
0020 150
0020 151 : SIDE EFFECTS:
0020 152
0020 153 : Signals: MTH$_INVARGMAT if |X| > 1 with reserved operand in R0/R1 (copied
0020 154 : to the signal mechanism vector CHF$MCH_R0/R1 by LIB$SIGNAL).
0020 155 : Associated message is: "INVALID ARGUMENT". Result is reserved operand -0.0
0020 156 : unless a user supplied (or any) error handler changes CHF$MCH_R0/R1.
0020 157
0020 158 : NOTE: This procedure disables floating point underflow, enables integer
0020 159 : overflow.
0020 160
0020 161 :---
0020 162
0020 163
0020 164 :.ENTRY MTH$GACOS, ^M<IV, R2, R3, R4, R5, R6, R7>
0022 165 : standard call-by-reference entry
0022 166 : disable DV (and FU), enable IV
0022 167 : flag that this is a jacket procedure in
0022
6D 00000000'GF 9E 0022 MOVAB G^MTH$$JACKET_HND, (FP)
```



```

; G Floating Point Arc-cosine routine
MTH$GACOS - Standard G Floating GACOS

```

Page 5
(4)

			0029			; set handler address to jacket
			0029			; handler
			0029			
			0029	168		; case of an error in routine
			0029	169		; If an error, convert signal to user PC
			0029	170		; and resignal
50	04	BC	50FD	0029	171	MOVG @value(AP), R0
		01	10	002E	172	BSBB MTH\$GACOS_R7
			04	0030	173	RET
				0031	174	

MT
1-


```
0031 176 .SBTTL MTH$GACOS_R7 - Special GACOS routine
0031 177
0031 178 ; Special GACOS - used by the standard routine and direct JSB call.
0031 179
0031 180 ; CALLING SEQUENCE:
0031 181 ; save anything needed in R0:R7
0031 182 ; MOVG R0 ; input in R0/R1
0031 183 ; JSB MTH$GACOS_R7
0031 184 ; RSB ; return with result in R0/R1
0031 185 ;
0031 186
0031 187 MTH$GACOS_R7:: ; special GACOS routine
0031 188 MTH$GACOS_R9:: ; Release 1 name
0031 189 ; save X in R6/R7
56 50 50FD 0031 189 MOVG R0, R6
05 12 0035 190 BNEQ TEST_FOR_1.0 ; branch if !X! > 0
0037 191
0037 192 ;
0037 193 ; X = 0
0037 194 ;
0037 195
50 C6 AF 7D 0037 196 MOVQ G_PI_OVER_2, R0 ; R0/R1 = PI/2
05 05 003B 197 RSB ; return PI/2 if !X! = 0
003C 198
003C 199 ;
003C 200 ; 0 < !X!
003C 201 ;
003C 202
003C 203 TEST_FOR_1.0:
50 8000 8F AA 003C 204 BICW #X8000, R0 ; R0/R1 = !X!
08 50 51FD 0041 205 CMPG R0, #1 ; compare !X! with 1.0
36 18 0045 206 BGEQ GEQ_TO_1.0 ; branch if !X! >= 1.0
0047 207
0047 208 ;
0047 209 ; 0 < !X! < 1.0
0047 210 ;
0047 211
50 4000 8F B1 0047 212 CMPW #X4000, R0 ; Check for possible loss of
0F 14 004C 213 BGTR 1$ ; significance
52 08 50 43FD 004E 214 SUBG3 R0, #1, R2 ; R2/R3 = 1 - X
50 08 40FD 0053 215 ADDG2 #1, R0 ; R0/R1 = 1 + X
50 52 44FD 0057 216 MULG2 R2, R0 ; R0/R1 = 1 - X^2
09 11 005B 217 BRB 2$ ; Join main flow
50 50 44FD 005D 218 1$: MULG2 R0, R0 ; R0/R1 = X**2
08 50 43FD 0061 219 SUBG3 R0, #1, R0 ; R0/R1 = 1.0 - X**2
00000000 GF 16 0066 220 2$: JSB G^MTH$GSQRT_R5 ; R0/R1 = GSQRT(1-X**2)
50 56 46FD 006C 221 DIVG2 R6, R0 ; R0/R1 = GSQRT(1-X**2)/X
56 DD 0070 222 PUSHL R6 ; save sign of X for sign test
00000000 GF 16 0072 223 JSB G^MTH$GATAN_R7 ; R0/R1 = GATAN(GSQRT(1-X**2)/X)
56 8E D0 0078 224 MOVL (SP)+, R6 ; restore sign of X
04 11 007B 225 BRB TEST_SIGN ; branch to TEST_SIGN
007D 226
007D 227 ;
007D 228 ; 1 <= !X!
007D 229 ;
007D 230
007D 231 GEQ_TO_1.0:
0E 14 007D 232 BGTR ERROR ; branch to ERROR if !X! > 1.0
```

```
007F 233
007F 234
007F 235 ;
007F 236 ; !X! = 1.0
007F 237 ;
50 7C 007F 238
0081 239 CLRQ R0 ; R0/R1 = 0
0081 240
0081 241
0081 242 ;
0081 243 ; Test the sign of X in order to decide if add PI to the result
0081 244 ;
0081 245
0081 246 TEST_SIGN:
56 53FD 0081 247 TSTG R6 ; test the sign of X
06 18 0084 248 BGEQ 10$ ; branch if X > 0
50 FF7D CF 40FD 0086 249 ADDG2 G_PI, R0 ; add PI to R0/R1 if X < 0
05 008C 250 10$: RSB ; return with result in R0/R1
008D 251
008D 252 ;
008D 253 ; 1 < !X!, error
008D 254 ;
008D 255
7E 00 6E DD 008D 256 ERROR: PUSHL (SP) ; return PC from JSB routine
50 01 0F 79 008F 257 MOVZBL #MTH$K_INVARGMAT, -(SP) ; condition value
0093 258 ASHQ #15, #T, R0 ; R0 = result = reserved operand -0.0
0097 259 ; goes to signal mechanism vector
0097 260 ; (CHF$L_MCH_R0/R1) so error handler
0097 261 ; can modify the result.
00000000'GF 02 FB 0097 262 CALLS #2, G^MTH$$SIGNAL ; signal error and use real user's PC
009E 263 ; independent of CALL vs JSB
05 009E 264 RSB ; return - R0 restored from CHF$L_MCH_R0/R1
009F 265
```



```
009F 267 .SBTTL MTH$GACOSD - Standard G Floating GACOSD
009F 268
009F 269
009F 270 :++
009F 271 : FUNCTIONAL DESCRIPTION:
009F 272 :
009F 273 : GACOSD - G floating point function
009F 274 :
009F 275 : GACOSD(X) is computed as:
009F 276 :
009F 277 :     If X = 0, then GACOSD(X) = 90.
009F 278 :     If X = 1, then GACOSD(X) = 0.
009F 279 :     If X = -1, then GACOSD(X) = 180.
009F 280 :     If 0 < X < 1, then GACOSD(X) = GATAND(SQRT(1-X**2)/X).
009F 281 :     If 1/2 =< X < 1, then GACOS(X) = GATAND(SQRT((1-X)*(1+X))/X).
009F 282 :     If -1 < X < 0, then GACOSD(X) = GATAND(SQRT(1-X**2)/X) + 180.
009F 283 :     If -1 < X =< -1/2, then GACOS(X) = GATAND(SQRT((1-X)*(1+X))/X) + 180.
009F 284 :     If 1 < |X|, error.
009F 285 :
009F 286 : CALLING SEQUENCE:
009F 287 :
009F 288 :     GACOSD.wg.v = MTH$GACOSD(x.rg.r)
009F 289 :
009F 290 : INPUT PARAMETERS:
009F 291 :
00000004 009F 292 :     LONG = 4 ; define longword multiplier
00000004 009F 293 :     x = 1 * LONG ; Contents of x is the argument
009F 294 :
009F 295 : IMPLICIT INPUTS: none
009F 296 :
009F 297 : OUTPUT PARAMETERS:
009F 298 :
009F 299 :     VALUE: G floating arc-cosine of the argument
009F 300 :
009F 301 : IMPLICIT OUTPUTS: none
009F 302 :
009F 303 : COMPLETION CODES: none
009F 304 :
009F 305 : SIDE EFFECTS:
009F 306 :
009F 307 : Signals: MTH$_INVARGMAT if |X| > 1 with reserved operand in R0/R1 (co180ed
009F 308 : to the signal mechanism vector CHFS$_MCH_R0/R1 by LIB$SIGNAL).
009F 309 : Associated message is: "INVALID ARGUMENT". Result is reserved operand -0.0
009F 310 : unless a user supplied (or any) error handler changes CHFS$_MCH_R0/R1.
009F 311 :
009F 312 : NOTE: This procedure disables floating point underflow, enables integer
009F 313 : overflow.
009F 314 :
009F 315 : ---
009F 316 :
40FC 009F 317 :
00A1 009F 318 : .ENTRY MTH$GACOSD, ^M<IV, R2, R3, R4, R5, R6, R7>
00A1 319 : ; standard call-by-reference entry
00A1 320 : ; disable DV (and FU), enable IV
00A1 321 : MTH$FLAG_JACKET ; flag that this is a jacket procedure in
6D 00000000'GF 9E 00A1 MOVAB G^MTH$$JACKET_HND, (FP)
```

```

; G Floating Point Arc-cosine routine      H 5
MTH$GACOSD - Standard G Floating GACOSD    16-SEP-1984 01:24:21 VAX/VMS Macro V04-00      Page 9
                                           6-SEP-1984 11:23:16 [MTHRTL.SRC]MTHGACOS.MAR;1      (6)

```

			00A8				; set handler address to jacket
			00A8				; handler
			00A8				
			00A8	322			; case of an error in routine
			00A8	323			; If an error, convert signal to user PC
			00A8	324			; and resignal
50	04	BC	50FD	00A8	325	MOVG	@value(AP), R0
		01	10	00AD	326	BSBB	MTH\$GACOSD_R7
			04	00AF	327	RET	; call special GACOSD routine
			00B0	328			; return with result in R0/R1


```
00B0 330      .SBTTL MTH$GACOSD_R7 - Special GACOSD routine
00B0 331
00B0 332      : Special GACOSD - used by the standard routine and direct JSB call.
00B0 333
00B0 334      CALLING SEQUENCE:
00B0 335      : save anything needed in R0:R7
00B0 336      : MOVG      R0      : input in R0/R1
00B0 337      : JSB      MTH$GACOSD_R7
00B0 338      : RSB      : return with result in R0/R1
00B0 339
00B0 340
00B0 341      MTH$GACOSD_R7::
00B0 342      : special GACOSD routine
00B0 343      : save X in R6/R7
00B0 344      : BNEQ      D_TEST_FOR_1.0      : branch if !X! > 0
00B0 345
00B0 346      : X = 0
00B0 347
00B0 348
00B0 349      : MOVQ      G_90, R0      : R0/R1 = 90
00B0 350      : RSB      : return 90 if !X! = 0
00B0 351
00B0 352      : 0 < !X!
00B0 353
00B0 354
00B0 355
00B0 356      D_TEST_FOR_1.0:
00B0 357      : BICW      #^X8000, R0      : R0/R1 = !X!
00B0 358      : CMPL      R0, #1      : compare !X! with 1.0
00B0 359      : BGEQ      D_GEQ_TO_1.0      : branch if !X! >= 1.0
00B0 360
00B0 361      : 0 < !X! < 1.0
00B0 362
00B0 363
00B0 364
00B0 365      : CMPW      #^X4000, R0      : Check for possible loss of
00B0 366      : BGTR      1$      : significance
00B0 367      : SUBG3      R0, #1, R2      : R2/R3 = 1 - X
00B0 368      : ADDG2      #1, R0      : R0/R1 = 1 + X
00B0 369      : MULG2      R2, R0      : R0/R1 = 1 - X^2
00B0 370      : BRB      2$      : Join main flow
00B0 371      1$:      : MULG2      R0, R0      : R0/R1 = X**2
00B0 372      : SUBG3      R0, #1, R0      : R0/R1 = 1.0 - X**2
00B0 373      2$:      : JSB      G^MTH$GSQRT_R5      : R0/R1 = GSQRT(1-X**2)
00B0 374      : DIVG2      R6, R0      : R0/R1 = GSQRT(1-X**2)/X
00B0 375      : PUSHL      R6      : save sign of X for sign test
00B0 376      : JSB      G^MTH$GATAND_R7      : R0/R1 = GATAND(GSQRT(1-X**2)/X)
00B0 377      : MOVL      (SP)+, R6      : restore sign of X
00B0 378      : BRB      D_TEST_SIGN      : branch to D_TEST_SIGN
00B0 379
00B0 380      : 1 <= !X!
00B0 381
00B0 382
00B0 383
00B0 384      D_GEQ_TO_1.0:
00B0 385      : BEQL      10$      : branch to ERROR if !X! > 1.0
00B0 386      : BRW      ERROR
```

```

0102 387
0102 388
0102 389 :
0102 390 : !X! = 1.0
0102 391 :
0102 392 :
50 7C 0102 393 10$: CLRQ R0 ; R0/R1 = 0
0104 394
0104 395
0104 396 :
0104 397 : Test the sign of X in order to decide if add 180 to the result
0104 398 :
0104 399 :
0104 400 D_TEST_SIGN:
0104 401 TSTG R6 ; test the sign of X
06 53FD 0107 402 BGEQ 10$ ; branch if X > 0
50 FFOA CF 40FD 0109 403 ADDG2 G_180, R0 ; add 180 to R0/R1 if X < 0
05 010F 404 10$: RSB ; return with result in R0/R1
0110 405
0110 406
0110 407
0110 408
0110 409 .END

```



```

D_GEQ TO 1.0      000000FD R    01
D_TEST_FOR 1.0    000000BC R    01
D_TEST_SIGN      00000104 R    01
ERROR            0000008D R    01
GEQ TO 1.0       0000007D R    01
G_180            00000018 R    01
G_90             00000010 R    01
G_PI             00000008 R    01
G_PI_OVER_2      00000000 R    01
LONG = 00000004
MTH$JACKET_HND ***** X    01
MTH$SIGNAL ***** X    00
MTH$GACOS        00000020 RG    01
MTH$GACOSD       0000009F RG    01
MTH$GACOSD R7    000000B0 RG    01
MTH$GACOS R7     00000031 RG    01
MTH$GACOS R9     00000031 RG    01
MTH$GATAN R7     ***** X    00
MTH$GATAN R7     ***** X    00
MTH$GSQRT R5     ***** X    00
MTH$K_INVARGMAT ***** X    00
TEST_FOR 1.0     0000003C R    01
TEST_SIGN        00000081 R    01
VALUE = 00000004

```

```

+-----+
! Psect synopsis !
+-----+

```

PSECT name	Allocation	PSECT No.	Attributes															
ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE					
_MTH\$CODE	00000110 (272.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG					

```

+-----+
! Performance indicators !
+-----+

```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	33	00:00:00.11	00:00:00.63
Command processing	111	00:00:00.62	00:00:03.39
Pass 1	92	00:00:01.16	00:00:03.64
Symbol table sort	0	00:00:00.01	00:00:00.01
Pass 2	79	00:00:01.00	00:00:03.87
Symbol table output	4	00:00:00.03	00:00:00.03
Psect synopsis output	2	00:00:00.05	00:00:00.23
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	323	00:00:02.98	00:00:11.86

The working set limit was 1050 pages.
6134 bytes (12 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 25 non-local and 7 local symbols.
469 source lines were read in Pass 1, producing 14 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.

+-----+
! Macro library statistics !
+-----+

Macro library name

Macros defined

_ \$255\$DUA28:[SYSLIB]STARLET.MLB;2

0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LI\$:MTHGACOS/OBJ=OBJ\$:MTHGACOS MSRC\$:MTHJACKET/UPDATE=(ENH\$:MTHJACKET)+MSRC

MT
SY
D-
ER
GE
G-
LO
MT
MT
MT
MT
MT
MT
MT
MT
RE
VA

PS
--
-M

Ph
--
In
Co
Pa
Sy
Pa
Sy
Ps
Cr
As

Th
51
Th
41
1

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY